

Bordeaux INP
ENSEIRB
MATMECA



Bureau d'étude DLD

Satellite de type Myriade

Encadré par David Henry

AUBARD Clément

BARROSO Stéphane

2016/2017 - S9

Groupe 1

Table des matières

INTRODUCTION	2
PARTIE N°1. Modèle linéarisé tangent	2
1.1 Détermination du modèle d'état non linéaire	2
1.2 Détermination du modèle linéarisé tangent	3
PARTIE N°2. Diagnostic d'un axe star tracker	4
2.1 Propriétés de détectabilité structurelle	4
2.2 Nouvelle expression des matrices	6
2.3 Propriétés d'observabilités	7
2.4 Vecteur de résidus	8
2.5 Jeu de données de simulation avec un défaut	9
2.6 Forme d'évaluation du résidu	9
2.7 Moyenne et variance de $r(k)$	10
2.8 Conclusion sur l'algorithme de diagnostic	10
PARTIE N°3. Diagnostic des tuyères	11
3.1 Propriétés de détectabilité structurelle	11
3.2 Nouvelle expression des matrices	11
3.3 Propriété d'observabilité	12
3.4 Estimateur de Kalman	12
3.5 Choix des matrices de covariance	13
3.6 Jeu de données de simulation avec un défaut	14
3.8 Moyenne et variance de $r(k)$	16
3.9 Conclusion sur l'algorithme de diagnostic	18
CONCLUSION	18
ANNEXES	18

INTRODUCTION

Le but de ce bureau d'étude va être de mettre en application les cours du semestre relatifs à la détection de défauts. Il faudra tout d'abord établir le modèle non linéaire puis le modèle linéarisé du satellite. Nous allons dans un premier temps, simuler un défaut sur le capteur θ , puis dans un second temps sur la tuyère n°2.

PARTIE N°1. Modèle linéarisé tangent

1.1 Détermination du modèle d'état non linéaire

Afin de déterminer le modèle d'état non linéaire, nous utilisons les relations fournies dans l'énoncé.

La relation liant les angles d'attitudes et les vitesses angulaires est la suivante :

$$\dot{\Theta}(t) = \frac{1}{\cos(\theta)} \begin{bmatrix} \cos(\theta) & \sin(\varphi) \sin(\theta) & \cos(\varphi) \sin(\theta) \\ 0 & \cos(\varphi) \cos(\theta) & -\sin(\varphi) \cos(\theta) \\ 0 & \sin(\varphi) & \cos(\varphi) \end{bmatrix} \omega(t)$$

Avec

$$\omega = (p \ q \ r)^T \quad \Theta = (\phi \ \theta \ \psi)^T$$

L'équation des moments est donnée par :

De plus nous prenons en compte les modes souples :

$$\ddot{\bar{q}} + 2\xi\omega_0\dot{\bar{q}} + \omega_0^2\bar{q} = -L^T\dot{\omega} \quad \bar{q} \in \mathbb{R}^{n_s \cdot n_p} \quad C_{sa} = -k \left(L\ddot{\bar{q}} + \sum_{i=1}^{n_p} J_{sai}\dot{\omega} \right)$$

$$\text{Avec } \bar{q} = (q_1 \ q_2)^T$$

Nous définissons donc les vecteurs suivants :

$$x = (\varphi \ \theta \ \psi \ p \ q \ r \ q_1 \ q_2 \ \dot{q}_1 \ \dot{q}_2)^T$$

$$y = (\varphi \ \theta \ \psi \ p \ q \ r)^T$$

$$u = C_{prop} = (C_{prop1} \ C_{prop2} \ C_{prop3})^T$$

$$d = C_{pert} = (C_{pert1} \ C_{pert2} \ C_{pert3})^T$$

Nous choisissons un modèle en moment pour respecter les notations de l'énoncé dans la partie diagnostic des tuyères.

1.2 Détermination du modèle linéarisé tangent

A l'aide de Matlab nous définissons le modèle linéarisé tangent autour de la position $x^* = 0$, $u^* = 0$ et $d^* = 0$.

Nous obtenons les matrices suivantes :

A =

0	0	0	1.0000	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0
0	0	0	0	0	1.0000	0	0	0	0
0	0	0	0	0	0	0.0029	0	0.0000	0
0	0	0	0	0	0	0	0.0001	0	0.0000
0	0	0	0	0	0	0	-0.0525	0	-0.0001
0	0	0	0	0	0	0	0	1.0000	0
0	0	0	0	0	0	0	0	0	1.0000
0	0	0	0	0	0	-1.1060	0	-0.0066	0
0	0	0	0	0	0	0	-26.9412	0	-0.0323

B =

0	0	0
0	0	0
0	0	0
0.0789	0	0
0	0.0769	-0.0000
0	-0.0000	0.0567
0	0	0
0	0	0
-2.8924	0	0
0	-0.0050	2.0980

C =

1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0

E1 =

0	0	0
0	0	0
0	0	0
0.0789	0	0
0	0.0769	-0.0000
0	-0.0000	0.0567
0	0	0
0	0	0
-2.8924	0	0
0	-0.0050	2.0980

E2 =

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

D =

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Nous utilisons la commande sigma SS afin de vérifier la cohérence du modèle, nous obtenons la figure suivant.

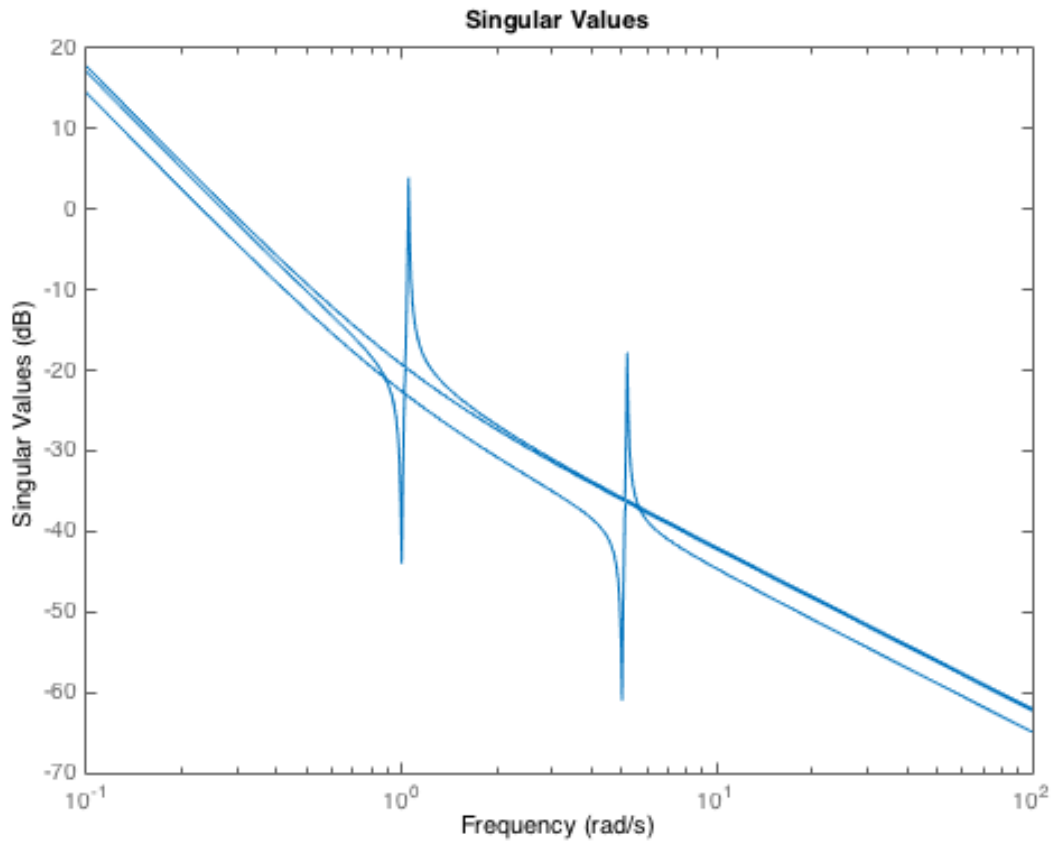


Figure 1 : Valeurs singulières du modèle linéarisé

Les pulsations $\omega=1\text{rd/s}$ et $\omega=5\text{rd/s}$ apparaissent sur le graphique, ce qui confirme la prise en compte du mode souple.

PARTIE N°2. Diagnostic d'un axe star tracker

2.1 Propriétés de détectabilité structurelle

Dans cette partie, nous appliquons un défaut sur le capteur θ .

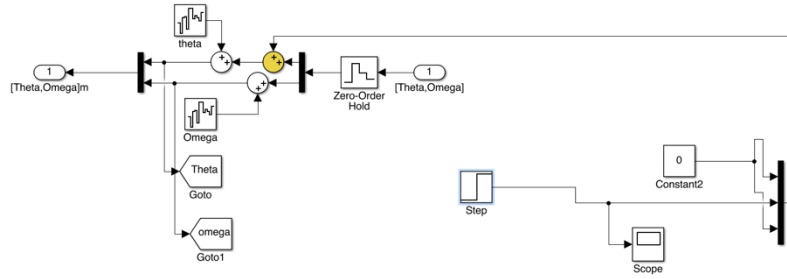


Figure 2 : Mise en place du défaut sur le capteur θ

Nous plaçons le défaut sur le capteur grâce au montage de la figure 2.

Afin de voir sur quelles variables influe le défaut, nous effectuons le schéma suivant.

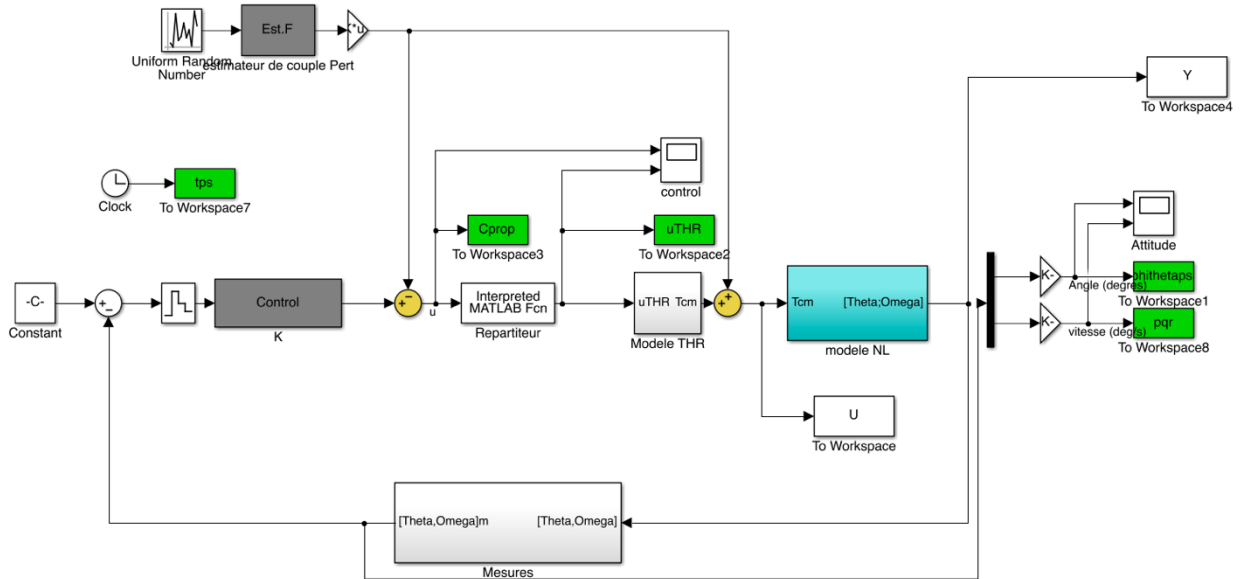


Figure 3 : Schéma permettant l'identification de l'influence du défaut

A l'aide du schéma de la figure 3, nous relevons les graphes du scope « Attitude » sans, puis avec défaut.

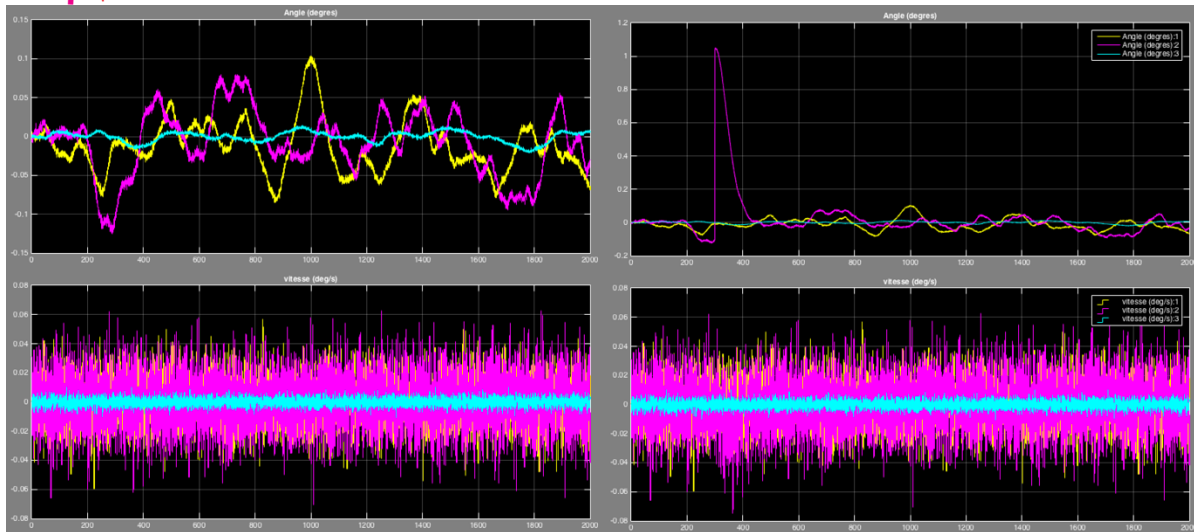


Figure 3 : Graphes des variables mesurées sans et avec défaut

La partie gauche de la figure est celle sans défaut et celle de droite est celle avec défaut.

Nous en déduisons donc que le défaut sur téta influe sur la position θ et la vitesse angulaire q .

2.2 Nouvelle expression des matrices

Grace à la partie 2.1 nous nous intéressons dans un premier temps uniquement aux variables θ et q et donc nous choisissons les matrices suivantes :

A =

0	0	0	1.0000	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0
0	0	0	0	0	1.0000	0	0	0	0
0	0	0	0	0	0	0.0029	0	0.0000	0
0	0	0	0	0	0	0	0.0001	0	0.0000
0	0	0	0	0	0	0	-0.0525	0	-0.0001
0	0	0	0	0	0	0	0	1.0000	0
0	0	0	0	0	0	0	0	0	1.0000
0	0	0	0	0	0	-1.1060	0	-0.0066	0
0	0	0	0	0	0	0	-26.9412	0	-0.0323

E1 =

B =

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0.0789	0	0	0.0789	0	0
0	0.0769	-0.0000	0	0.0769	-0.0000
0	-0.0000	0.0567	0	-0.0000	0.0567
0	0	0	0	0	0
0	0	0	0	0	0
-2.8924	0	0	-2.8924	0	0
0	-0.0050	2.0980	0	-0.0050	2.0980

K2 =

1
0

C =

0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0

D =

0	0	0
0	0	0

E2 =

0	0	0
0	0	0

K1=zeros(10,1)

2.3 Propriétés d'observabilités

Avec les matrices précédentes, le rang de la matrice AC est inférieur à 10 , elle n'est pas observable.

On choisit donc de prendre les variables suivantes φ θ ψ et q pour notre algorithme de détection.

Afin de créer les matrices φ_1 , φ_2 et φ_3 nous devons déterminer s, qui est défini comme suit :
 $m^*(S+1) > \text{rang}(OS)$.

Pour cela on calcule rank (Obs)=9 avec Obs=[C;C*A;C*A*A] de plus nous avons 4 variables de mesures donc nous en déduisons s= 2.

Nous devons recalculer les matrices pour avoir les 4 variables choisies.

A =

0	0	0	1.0000	0	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0	0
0	0	0	0	0	1.0000	0	0	0	0	0
0	0	0	0	0	0	0.0029	0	0.0000	0	0
0	0	0	0	0	0	0	0.0001	0	0.0000	0
0	0	0	0	0	0	0	-0.0525	0	-0.0001	0
0	0	0	0	0	0	0	0	1.0000	0	0
0	0	0	0	0	0	0	0	0	1.0000	0
0	0	0	0	0	0	-1.1060	0	-0.0066	0	0
0	0	0	0	0	0	0	-26.9412	0	-0.0323	0

C =

1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0

D =

0	0	0
0	0	0
0	0	0
0	0	0

E2 =

0	0	0
0	0	0
0	0	0
0	0	0

K2 =

0
1
0
0

B =

0	0	0
0	0	0
0	0	0
0.0789	0	0
0	0.0769	-0.0000
0	-0.0000	0.0567
0	0	0
0	0	0
-2.8924	0	0
0	-0.0050	2.0980

E1 =

0	0	0
0	0	0
0	0	0
0.0789	0	0
0	0.0769	-0.0000
0	-0.0000	0.0567
0	0	0
0	0	0
-2.8924	0	0
0	-0.0050	2.0980

2.4 Vecteur de résidus

Nous appliquons la méthode de l'espace de parité dynamique pour construire un vecteur de résidus.

V =

-1.0000
0.8843
-0.0027

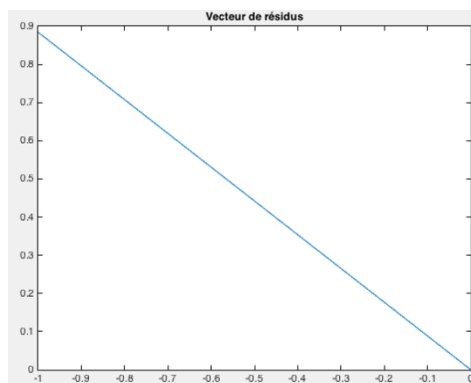


Figure 4 : Vecteur de résidus

Nous constatons que 2 valeurs du vecteur sont assez élevées. Ce qui correspond aux variables θ et q .

2.5 Jeu de données de simulation avec un défaut

Le défaut choisi est un échelon d'amplitude 0.02 à partir de $t=300$.

A l'aide du simulateur, nous relevons un jeu de données de simulation.

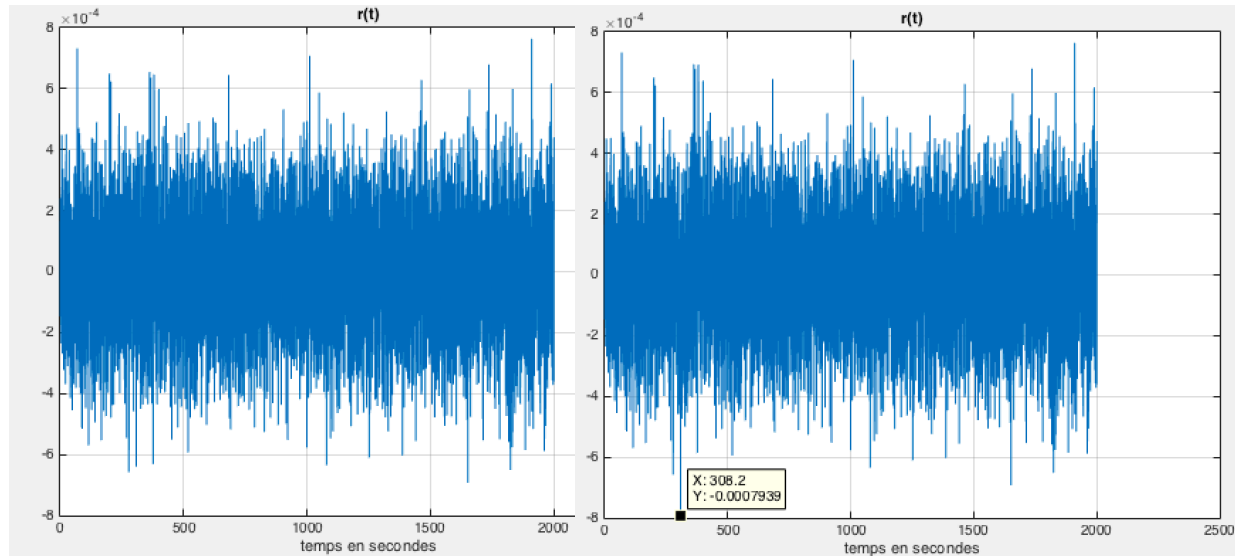


Figure 5 : Tracé de $r(t)$ sans et avec défaut

A gauche nous avons $r(t)$ sans défaut et à droite avec défaut. Nous pouvons voir une légère modification sur $r(t)$ avec le défaut, le défaut est détecté à $t=308.2s$.

L'amplitude du défaut choisie étant faible, la mise en évidence de ce dernier n'est pas évidente sur cette figure.

2.6 Forme d'évaluation du résidu

La forme d'évaluation du résidu nous donne l'allure de ce que nous voulons obtenir en simulation et doit être cohérente avec la partie 2.5.

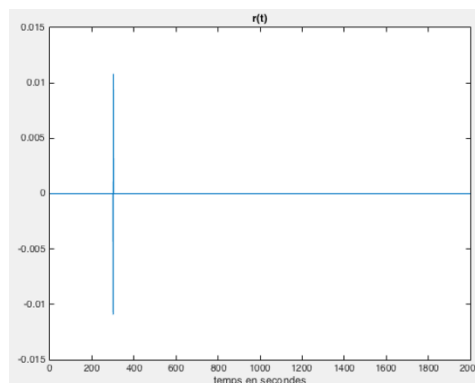


Figure 6 : Forme d'évaluation du résidu

Le défaut apparaît sur cette figure à $t=300s$. Cette figure semble être une représentation plus nette de la figure 4 (A l'amplitude près), ce qui est en accord avec la simulation.

2.7 Moyenne et variance de $r(k)$

Nous appliquons dans cette partie un test séquentiel sur le changement de moyenne et de variance de $r(k)$.

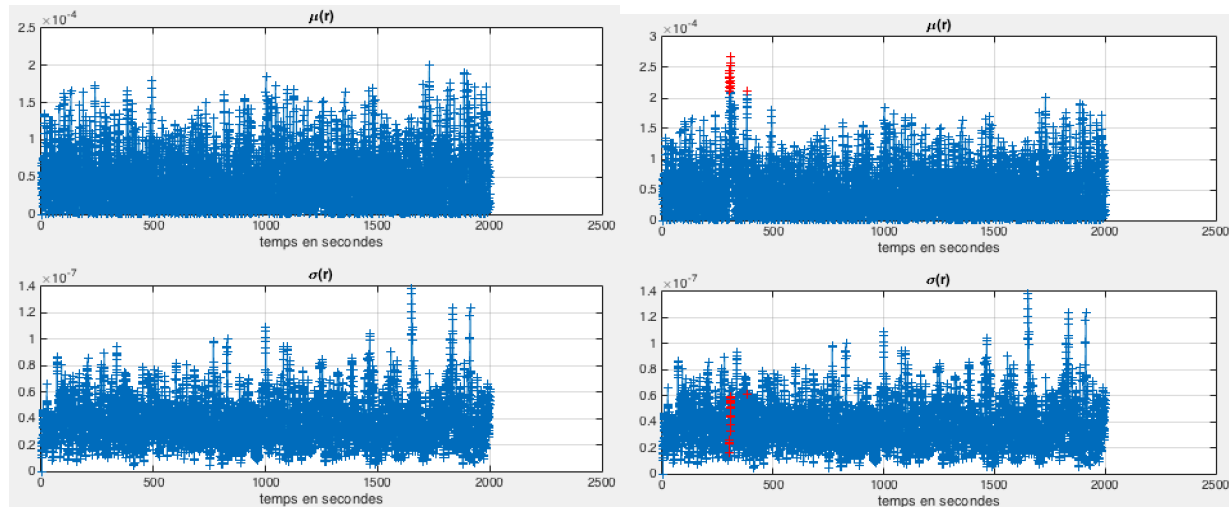


Figure 7 : Changement de moyenne et de variance de $r(k)$ sans et avec défaut

A gauche nous avons aucun défaut et à droite nous avons notre défaut. Nous pouvons voir sur la droite de la figure, deux détections de défaut, notre défaut est détecté à $t=308.2s$.

Nous pouvons aussi remarquer une fausse alarme à 400s.

2.8 Conclusion sur l'algorithme de diagnostic

Notre algorithme détecte le moment d'apparition du défaut avec 8 secondes de retard. Nous avons cependant une fausse alarme.

Après plusieurs itérations de défauts, il s'avère que notre algorithme ne détecte pas un défaut de type échelon inférieur à 0.013.

PARTIE N°3. Diagnostic des tuyères

3.1 Propriétés de détectabilité structurelle

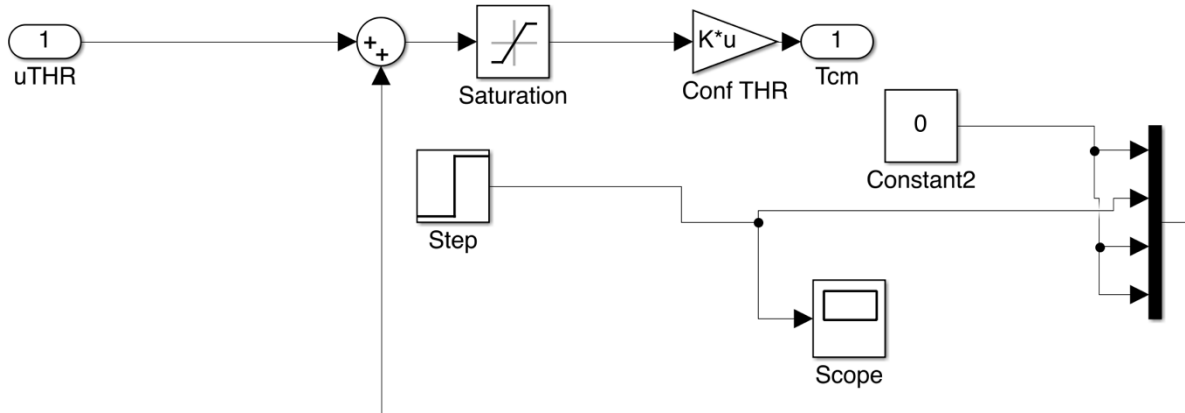


Figure 8 : Mise en place du défaut sur la deuxième tuyère

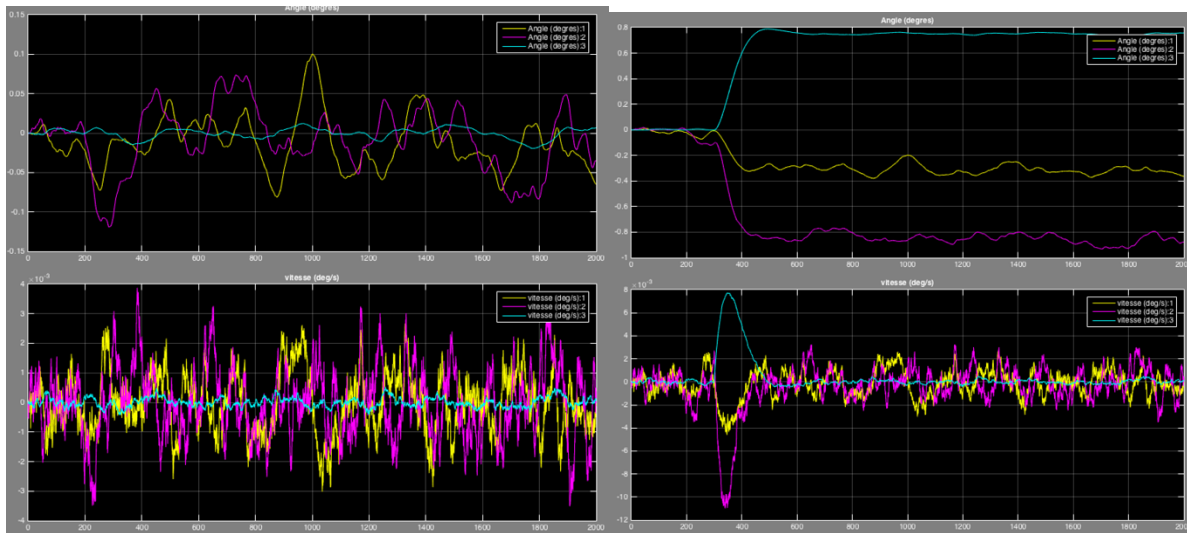


Figure 9 : Graphes des variables de sorties sans et avec défaut

La partie gauche de la figure correspond à la sortie du satellite sans le défaut de tuyère et la partie droite avec le défaut de tuyère.

Nous en déduisons que le défaut sur la tuyère 2 influe sur les variables $\varphi, \theta, \psi, p, q$ et r .

3.2 Nouvelle expression des matrices

Grâce à la partie précédente, nous choisissons les matrices suivantes :

A =

0	0	0	1.0000	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0
0	0	0	0	0	1.0000	0	0	0	0
0	0	0	0	0	0	0.0029	0	0.0000	0
0	0	0	0	0	0	0	0.0001	0	0.0000
0	0	0	0	0	0	0	-0.0525	0	-0.0001
0	0	0	0	0	0	0	0	1.0000	0
0	0	0	0	0	0	0	0	0	1.0000
0	0	0	0	0	0	-1.1060	0	-0.0066	0
0	0	0	0	0	0	0	-26.9412	0	-0.0323

B =

0	0	0
0	0	0
0	0	0
0.0789	0	0
0	0.0769	-0.0000
0	-0.0000	0.0567
0	0	0
0	0	0
-2.8924	0	0
0	-0.0050	2.0980

C =

1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0

D =

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

E2 =

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

K1=zeros(10,1) et K2=zeros(6,1).

3.3 Propriété d'observabilité

Nous avons $\text{rank}(\text{obsv}(AC))=10$, donc nous ne modifions pas les matrices précédentes.

3.4 Estimateur de Kalman

Dans cette partie, on décide d'utiliser une approche de type "estimateur minimisant la variance de l'erreur d'estimation". La solution cherchée est donc basée sur l'estimateur de Kalman. Donner la représentation d'état continue étendue permettant de construire $\hat{f}(t)$, l'estimée du défaut $f(t)$, sous la forme suivante :

$$\begin{cases} \dot{x}_e = A_e x_e + B_e u + G w \\ y_e = C_e x_e + D_e u + H w + v \end{cases}$$

Nous définissons les matrices suivantes :

Ae =

0	0	0	1.0000	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0
0	0	0	0	0	1.0000	0	0	0	0
0	0	0	0	0	0	0.0029	0	0.0000	0
0	0	0	0	0	0	0	0.0001	0	0.0000
0	0	0	0	0	0	0	-0.0525	0	-0.0001
0	0	0	0	0	0	0	0	1.0000	0
0	0	0	0	0	0	0	0	0	1.0000
0	0	0	0	0	0	-1.1060	0	-0.0066	0
0	0	0	0	0	0	0	-26.9412	0	-0.0323

Be =

0	0	0
0	0	0
0	0	0
0.0789	0	0
0	0.0769	-0.0000
0	-0.0000	0.0567
0	0	0
0	0	0
-2.8924	0	0
0	-0.0050	2.0980

Ce =

1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0

De =

0	0	0			
0	0	0			
0	0	0			
0	0	0			
0	0	0			
0	0	0			
			0	0	0
			0	0	0
			0	0	0
			0.0789	0	0
			0	0.0769	-0.0000
			0	-0.0000	0.0567
			0	0	0
			0	0	0
			-2.8924	0	0
			0	-0.0050	2.0980

3.5 Choix des matrices de covariance

Le choix des matrices de covariance est la partie la plus compliquée dans cette approche, en plus de devoir gérer les dimensions des matrices, il a fallu respecter le fait que certains produits de

matrices soient semi-définis positifs. C'est pour cela que nous avons $N \neq 0$. Et donc pas un estimateur de Kalman optimal.

$$Q = \begin{matrix} & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{matrix} \quad N = 1e-10 * \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

1000000

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$L = 1.0e+03 * \begin{bmatrix} 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 \\ -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\ -0.0000 & -0.0000 & 0.0010 & -0.0000 & -0.0000 & 0.0010 \\ 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 \\ -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\ 0.0000 & -0.0000 & 0.0010 & -0.0000 & -0.0000 & 0.0567 \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 \\ -0.0000 & -0.0001 & 0.0014 & 0.0000 & -0.0000 & 0.0207 \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 \\ 0.0000 & -0.0000 & 0.0010 & 0.0000 & -0.0024 & 2.0874 \end{bmatrix}$$

$$P = 1.0e+05 * \begin{bmatrix} 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 \\ -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & -0.0000 \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\ -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & -0.0000 \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0006 & -0.0000 & 0.0002 & 0.0000 & 0.0209 \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & -0.0000 \\ -0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0002 & 0.0000 & 0.1745 & 0.0000 & 0.0021 \\ -0.0000 & -0.0000 & 0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\ 0.0000 & -0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0209 & -0.0000 & 0.0021 & 0.0000 & 5.1337 \end{bmatrix}$$

3.6 Jeu de données de simulation avec un défaut

Dans cette partie nous effectuons un défaut sur la tuyère n°2 (échelon d'amplitude 0.001 à partir de 300s)

A l'aide du simulateur, nous relevons un jeu de données de simulation.

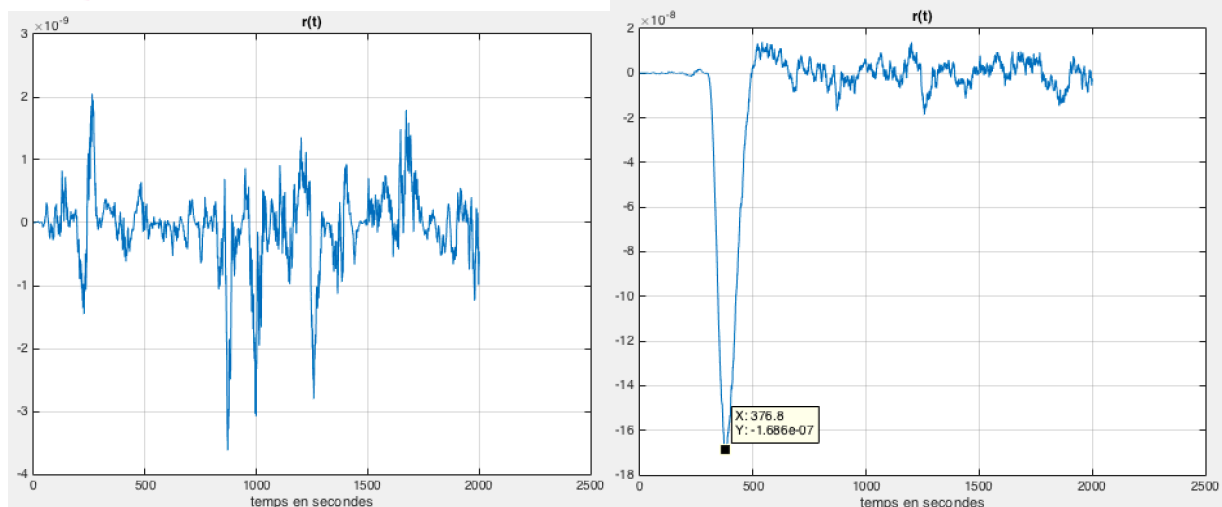


Figure 10 : Tracé de $r(t)$ sans et avec défaut

A gauche nous avons $r(t)$ sans défaut et à droite avec défaut. Nous pouvons voir une modification sur $r(t)$ avec le défaut.

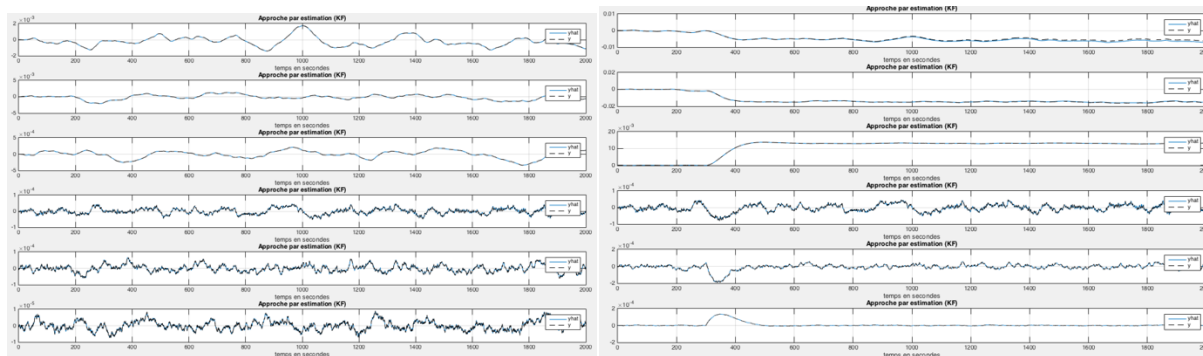


Figure 11 : Sortie et estimation de sortie sans et avec défaut

Nous constatons que notre estimateur est assez performant car les sorties estimées sont presque semblables aux sorties réelles. Nous voyons que le défaut est détecté par l'estimateur au moment où il apparaît.

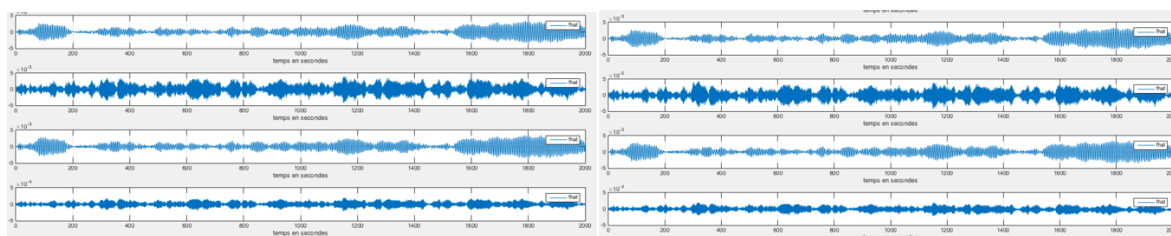


Figure 12 : Tracé de $\hat{f}(t)$

Sur cette figure nous avons l'estimé de $f(t)$ sans et avec défaut. Le défaut ne se voit pas directement, nous allons donc zoomé pour le voir plus nettement.

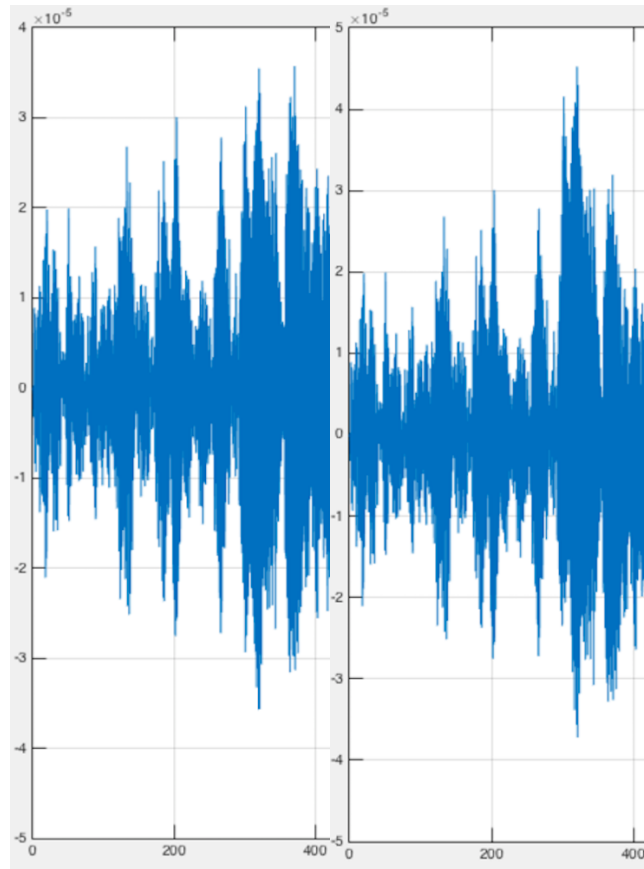


Figure 13 : Zoom sur $\hat{f}at2$

Nous effectuons un zoom sur $\hat{f}(t)2$ et nous constatons que le défaut est apparent à partir de 300.8s.

3.8 Moyenne et variance de $r(k)$

Nous appliquons un test séquentiel sur le changement de moyenne et de variance de $r(k)$.

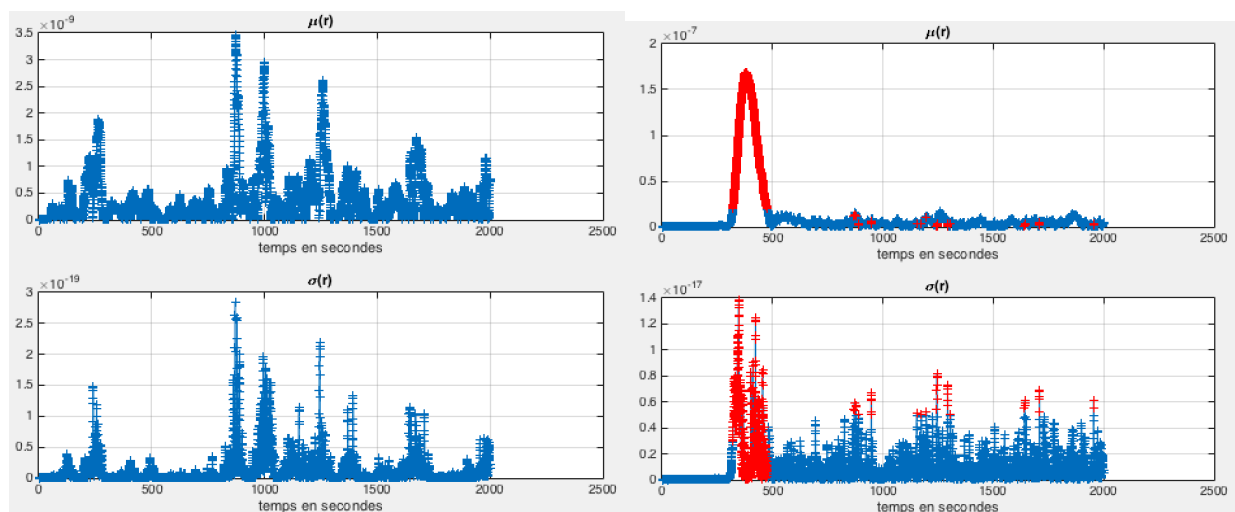


Figure 14 : Changement de moyenne et de variance de $r(k)$ sans et avec défaut de la méthode précédente

A gauche nous avons aucun défaut et à droite nous avons notre défaut.

Nous constatons que l'approche utilisée dans la partie 2 n'est pas adéquate car il y a de fausses alertes.

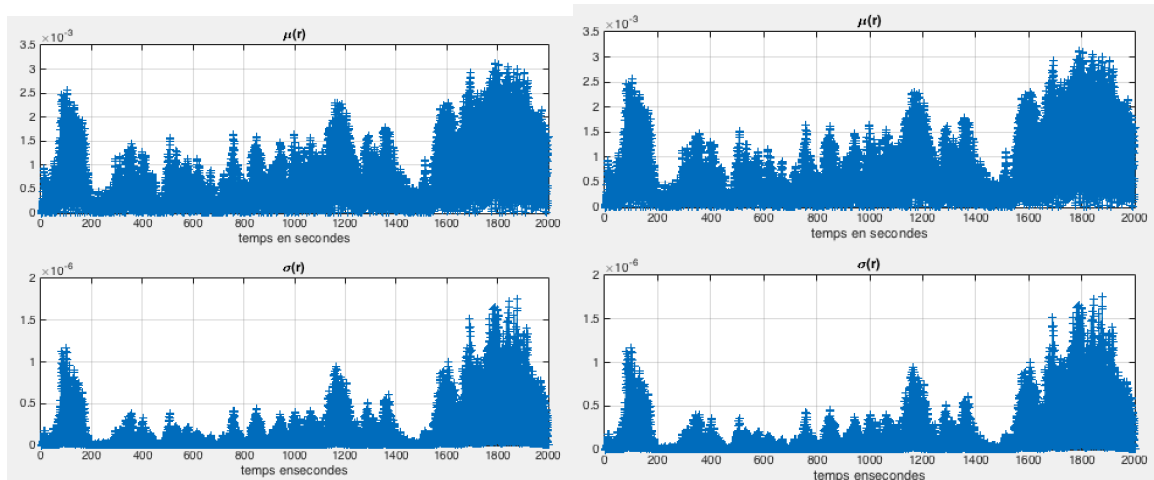


Figure 14 : Changement de moyenne et de variance de $r(k)$ sans et avec défaut avec l'approche de Kalman

A gauche nous avons aucun défaut et à droite nous avons notre défaut. Le défaut n'apparaît pas et il n'y a pas de fausses alertes. Le défaut n'est pas détecté. L'amplitude est peut-être trop faible ou le f estimé est peut-être faux.

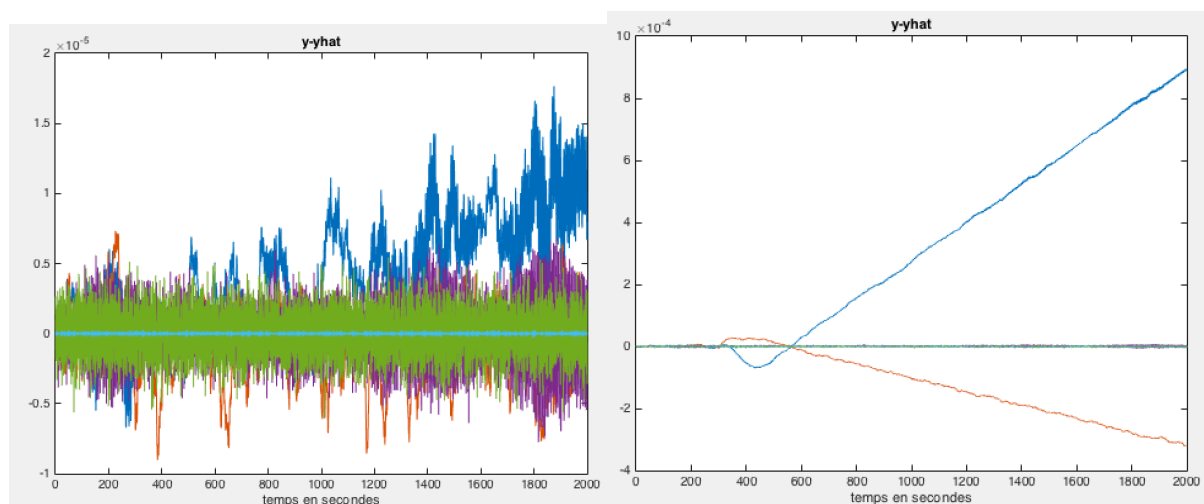


Figure 15 : Changement de moyenne et de variance de $r(k)$ sans et avec défaut avec l'approche de Kalman

Nous remarquons que l'estimateur n'est pas performant sur 2 variables (courbes bleue et orange). Cela est sûrement dû à notre choix de matrice N.

Cependant si le filtre est bien réalisé, une approche par résidus n'est pas utile car il n'y aura aucune variation entre y et \hat{y} .

3.9 Conclusion sur l'algorithme de diagnostic

Le retard à la détection est de 0.8s.

Il n'y a ni fausse alerte ni non détection de défaut.

CONCLUSION

Ce bureau d'étude nous a permis d'acquérir des connaissances sur la détection de défaut et sur l'approche par estimateur de Kalman. Nous avons essayé de faire au mieux pour obtenir des résultats cohérents. Cependant, il n'est pas évident de traiter un système sur lequel nous avons très peu de notions pratiques (Satellite) et sur l'application directe du cours sans avoir fait au préalable un TP sur le sujet. Ce bureau d'étude n'est donc pas un franc succès et nous allons avoir besoin de plus de pratique afin de maîtriser ces notions.

ANNEXES

```
%capteur Teta
%% Modele NL
clear all
close all
clc
global Torque
load('dataForSat.mat');
load('dataFlexibleModes.mat')
load('dataK.mat')
InertieSat=diag([14 13 19]);

Te=1/4;
umax=1;umin=0;
Est.F=c2d(Est.F,Te,'tustin');
numTHRdef=0;

%% Configuration THR
M = Torque.RepMat;
invM=pinv(M);Torque.pinvRepMat=invM;
umax=1; %N

load('dataFlexibleModes.mat');
```

```
%% Constantes
Is=diag([14 13 19]);
epsi=diag(flex.ksi);
Jo=flex.inertie0;
wo=diag(flex.omega);
L=flex.BR;
K=0.001;

%% Variables
syms p q r phi teta psi Cprop1 Cprop2 Cprop3 Cpert1 Cpert2 Cpert3 Q1 Q2
Qdot1 Qdot2 real
Q=[Q1;Q2];
Qdot=[Qdot1;Qdot2];
Cprop=[Cprop1;Cprop2;Cprop3];
Cpert=[Cpert1;Cpert2;Cpert3];
W=[p;q;r];
TETA=[phi;teta;psi];

%% Vecteurs d'État/commande/sortie
u=Cprop;
d=Cpert;
TETAdot=1/cos(teta)*([cos(teta) sin(phi)*sin(teta) cos(phi)*sin(teta)]
                    [0 cos(phi)*cos(teta) -sin(phi)*cos(teta)]
                    [0 sin(phi) cos(phi)])*W;

%% Prise en compte du mode souple
Wdot=inv(-K*L*L'+K*Jo+Is)*(K*2*L*epsi*wo*Qdot+K*L*wo*wo*Q-
cross(W,Is*W)+Cprop+Cpert);
x=[TETA;W;Q;Qdot];
Qdotdot=-L'*Wdot-2*epsi*wo*Qdot-wo*wo*Q;
xdot=[TETAdot;Wdot;Qdot;Qdotdot];
y=[TETA;W];
%% Linéarisation

K1=zeros(10,1);

K2=[0 1 0 0 0 0]';
f=xdot;
g=y;
A=jacobian(f,x);B=jacobian(f,u);C=jacobian(g,x); D=jacobian(g,u);
E1=jacobian(f,d); E2=jacobian(g,d);

phi=0;teta=0;psi=0;p=0;q=0;r=0;Cprop1=0;Cprop2=0;Cprop3=0;Cpert1=0;Cper
t2=0;Cpert3=0;Q1=0;Q2=0;Qdot1=0;Qdot2=0;

A=eval(A);
B=eval(B);
C=eval(C);
D=eval(D);
E1=eval(E1);
E2=eval(E2);
```

```

K2=[0 1 0 0]';
C=C([1,2,3,5],:);

D=D([1,2,3,5],:)
E2=E2([1,2,3,5],:)
Rank=rank(observ(A,C))

%% Espace de parité
% Discretisation  $\Delta t$  250ms
Obs=[C;C*A;C*A*A];
rank(Obs)
W=(null(Obs'))'
% s=2
K1=zeros(10,1);
Te=0.25;
[Ad,BEKd,Cd,DEKd]=c2dm(A,[B E1 K1],C,[D E2 K2],Te,'zoh');
Bd=BEKd(:,1:3);E1d=BEKd(:,4:6);K1d=BEKd(:,7);
Dd=DEKd(:,1:3);E2d=DEKd(:,4:6);K2d=DEKd(:,7);
s=2;Os=[Cd;Cd*Ad;Cd*Ad*Ad];
W=null(Os')';
Phi1=[ [Dd;Cd*Bd;Cd*Ad*Bd],[zeros(size(Dd));Dd;Cd*Bd],[zeros(size(Dd));zeros(size(Dd));Dd]];
Phi2=[ [E2d;Cd*E1d;Cd*Ad*E1d],[zeros(size(E2d));E2d;Cd*E1d],[zeros(size(E2d));zeros(size(E2d));E2d]];
Phi3=[ [K2d;Cd*K1d;Cd*Ad*K1d],[zeros(size(K2d));K2d;Cd*K1d],[zeros(size(K2d));zeros(size(K2d));K2d]];
M1=W*Phi2*Phi2'*W';
M2=W*Phi3*Phi3'*W';
[vecteurp,valeurp]=eig(M1,M2);
idx=1;
v=vecteurp(:,idx)
W0=v'*W;
figure
plotv(v);
title('Vecteur de résidus');
N=2000;
n=5e-2*randn(3,N+1);
f=0.02*[zeros(1,300) ones(1,N-299)];
for k=s+1:N;
test(k)=W0*(Phi2*[n(:,k-2);n(:,k-1);n(:,k)]-Phi3*[f(:,k-2);f(:,k-1);f(:,k)]);
end
figure
plot(test)
title('r(t)');xlabel('temps en secondes')
%% Jeu de données (obtenu du simulateur)
sim('SimulatorSimplifie');y=[Y(:,1),Y(:,2),Y(:,3),Y(:,5)]';u=[U]';
for k=s+1:length(y);
ropt(k)=W0*([y(:,k-2);y(:,k-1);y(:,k)]-Phi3*[u(:,k)]);
end

t=(1:length(y)).*Te;
figure;plot(t,ropt);grid;title('r(t)');xlabel('temps en secondes')

```

```
% test moyenne/variance glissante
taillefen=10;
signaltoproceed=ropt;
for k=taillefen+1:length(signaltoproceed);
moyenne(k)=abs(mean(signaltoproceed(k-taillefen:k)));
variance(k)=abs(cov(signaltoproceed(k-taillefen:k)));
end
idx=find(moyenne>(2.1e-4) | variance>(1.4e-7)); %% on cherche quand la
moyenne > 7e-11 OU variance > 1.2e-20
%%Bode
%%
figure;
subplot(211);plot(t,moyenne,'-+');grid;title('\mu(r)'); hold
on;plot(t(idx),moyenne(idx),'r+');xlabel('temps en secondes')
subplot(212);plot(t,variance,'-+');grid;title('\sigma(r)'); hold
on;plot(t(idx),variance(idx),'r+');xlabel('temps en secondes')

%tuyere n $\infty$ 2
%% Modele NL
clear all
close all
clc
global Torque
load('dataForSat.mat');
load('dataFlexibleModes.mat');
load('dataK.mat');
InertieSat=diag([14 13 19]);
Te=1/4;
umax=1;umin=0;
Est.F=c2d(Est.F,Te,'tustin');
numTHRdef=0;
%% Configuration THR
M = Torque.RepMat;
invM=pinv(M);Torque.pinvRepMat=invM;
umax=1; %N
load('dataFlexibleModes.mat');
%% Constantes
Is=diag([14 13 19]);
epsi=diag(flex.ksi);
Jo=flex.inertie0;
wo=diag(flex.omega);
L=flex.BR;
K=0.001;
%% Variables
syms p q r phi teta psi Cprop1 Cprop2 Cprop3 Cpert1 Cpert2 Cpert3 Q1 Q2
Qdot1 Qdot2 real
Q=[Q1;Q2];
Qdot=[Qdot1;Qdot2];
Cprop=[Cprop1;Cprop2;Cprop3];
Cpert=[Cpert1;Cpert2;Cpert3];
W=[p;q;r];
```

```

TETA=[phi;teta;psi];

%% Vecteurs d'État/commande/sortie
u=Cprop;
d=Cpert;
TETAdot=1/cos(teta)*([cos(teta) sin(phi)*sin(teta) cos(phi)*sin(teta)]
                    [0 cos(phi)*cos(teta) -sin(phi)*cos(teta)]
                    [0 sin(phi) cos(phi)]])*W;

%% Prise en compte du mode souple
Wdot=inv(-K*L*L'+K*Jo+Is)*(K*2*L*epsi*wo*Qdot+K*L*wo*wo*Q-
cross(W,Is*W)+Cprop+Cpert);
x=[TETA;W;Q;Qdot];
Qdotdot=-L'*Wdot-2*epsi*wo*Qdot-wo*wo*Q;
xdot=[TETAdot;Wdot;Qdot;Qdotdot];
y=[TETA;W];
%% Linéarisation
f=xdot;
g=y;
A=jacobian(f,x);B=jacobian(f,u);C=jacobian(g,x); D=jacobian(g,u);
E1=jacobian(f,d); E2=jacobian(g,d);
phi=0;teta=0;psi=0;p=0;q=0;r=0;Cprop1=0;Cprop2=0;Cprop3=0;Cpert1=0;Cper
t2=0;Cpert3=0;Q1=0;Q2=0;Qdot1=0;Qdot2=0;
A=eval(A);
B=eval(B);
C=eval(C);
D=eval(D);
E1=eval(E1);
E2=eval(E2);
K1=zeros(10,1);
K2=[0 0 0 0 0 0]';
rank(observ(A,C));
C=C(1:6,:);
D=D(1:6,:);
E2=E2(1:6,:);
% point de linearisation, valeurs nominales et calcul des matrices d
etat
rank(observ(A,C));
%% Espace de parité
% Discretisation + 100m
Obs=[C;C*A;C*A*A];
rank(Obs);
W=(null(Obs'))';
% s=2
K1=zeros(10,1);
Te=0.25;
[Ad,BEKd,Cd,DEKd]=c2dm(A,[B E1 K1],C,[D E2 K2],Te,'zoh');
Bd=BEKd(:,1:3);E1d=BEKd(:,4:6);K1d=BEKd(:,7);
Dd=DEKd(:,1:3);E2d=DEKd(:,4:6);K2d=DEKd(:,7);
s=2;Os=[Cd;Cd*Ad;Cd*Ad*Ad];
W=null(Os')';
Phi1=[Dd;Cd*Bd;Cd*Ad*Bd],[zeros(size(Dd));Dd;Cd*Bd],[zeros(size(Dd));z
eros(size(Dd));Dd]];
Phi2=[E2d;Cd*E1d;Cd*Ad*E1d],[zeros(size(E2d));E2d;Cd*E1d],[zeros(size(

```

```

E2d));zeros(size(E2d));E2d]];
Phi3=[[K2d;Cd*K1d;Cd*Ad*K1d],[zeros(size(K2d));K2d;Cd*K1d],[zeros(size(
K2d));zeros(size(K2d));K2d]];
M1=W*Phi2*Phi2'*W';
M2=W*Phi3*Phi3'*W';
[vecteurp,valeurp]=eig(M1,M2);
idx=1;
v=vecteurp(:,idx);
W0=v'*W;
N=2000;
%% Jeu de données (obtenu du simulateur)
sim('TSimulatorSimplifie');y=Y;u=U;
u2=u';
y2=y';
for k=s+1:length(y);
ropt(k)=W0*([y2(:,k-2);y2(:,k-1);y2(:,k)]-Phi3*[u2(:,k)]);
end
t=(1:length(y)).*Te;
figure;plot(t,ropt);grid;title('r(t)');xlabel('temps en secondes')
% test moyenne/variance glissante
taillefen=10;
signaltoproceed=ropt;
for k=taillefen+1:length(signaltoproceed);
moyenne(k)=abs(mean(signaltoproceed(k-taillefen:k)));
variance(k)=abs(cov(signaltoproceed(k-taillefen:k)));
end
idx=find(moyenne>(2e-8) | variance>(0.5e-17)); %% on cherche quand la
moyenne > 7e-11 OU variance > 1.2e-20
figure;
subplot(211);plot(t,moyenne,'-+');grid;title('\mu(r)'); hold
on;plot(t(idx),moyenne(idx),'r+');xlabel('temps en secondes')
subplot(212);plot(t,variance,'-+');grid;title('\sigma(r)'); hold
on;plot(t(idx),variance(idx),'r+');xlabel('temps en secondes')
%% Diagnostic Estimateur de Kalman
Ae=[A];
Be=[B];
G=[E1] ;
Ce=[C];
De=D;
H=zeros(6,3);
Q=1e6
R=diag([1 1 1 1 1 1]);
N=1e-10*[1 1 0 0 0 0]; % un bon kalman c'est y-yhat = 0
[ssKalman,L,P] = kalman(ss(Ae,[Be G],Ce,[De H]),Q,R,N);
L;
P;
ssKalmand=c2d(ssKalman,Te,'tustin');
yxe = lsim(ssKalmand,[u,zeros(2,8001)'],y);
t=tps;
yhat(:,1)=yxe(:,1);
yhat(:,2)=yxe(:,2);
yhat(:,3)=yxe(:,3);
yhat(:,4)=yxe(:,4);%
```



```

yhat(:,5)=yxe(:,5);%
yhat(:,6)=yxe(:,6);
fhat=yxe(:,13:16);
figure(3);
subplot(611);plot(t,yhat(:,1));hold on;plot(t,y(:,1),'k--');
xlabel('temps en secondes');grid on;legend('yhat','y');
title('Approche par estimation (KF)')
subplot(612);plot(t,yhat(:,2));hold on;plot(t,y(:,2),'k--');
xlabel('temps en secondes');grid on;legend('yhat','y');
title('Approche par estimation (KF)')
subplot(613);plot(t,yhat(:,3));hold on;plot(t,y(:,3),'k--');
xlabel('temps en secondes');grid on;legend('yhat','y');
title('Approche par estimation (KF)')
subplot(614);plot(t,yhat(:,4));hold on;plot(t,y(:,4),'k--');
xlabel('temps en secondes');grid on;legend('yhat','y');
title('Approche par estimation (KF)')
subplot(615);plot(t,yhat(:,5));hold on;plot(t,y(:,5),'k--');
xlabel('temps en secondes');grid on;legend('yhat','y');
title('Approche par estimation (KF)')
subplot(616);plot(t,yhat(:,6));hold on;plot(t,y(:,6),'k--');
xlabel('temps en secondes');grid on;legend('yhat','y');
title('Approche par estimation (KF)')
figure(4)
subplot(411);plot(t,fhat(:,1));xlabel('temps en secondes'); grid
on;legend('fhat');
subplot(412);plot(t,fhat(:,2));xlabel('temps en secondes'); grid
on;legend('fhat');
subplot(413);plot(t,fhat(:,3));xlabel('temps en secondes'); grid
on;legend('fhat');
subplot(414);plot(t,fhat(:,4));xlabel('temps en secondes'); grid
on;legend('fhat');
taillefen=4;
signaltoproceed=fhat;
for k=taillefen+1:length(signaltoproceed);
moyenne(k)=abs(mean(signaltoproceed(k-taillefen:k)));
variance(k)=abs(cov(signaltoproceed(k-taillefen:k)));
end
idx=find(moyenne>3.5e-3 | variance>2e-6); %% on cherche quand la
moyenne > 3e-4 OU variance > 3e-10
figure(5);
subplot(211);plot(t,moyenne,'-+');grid;title('\mu(r)'); hold
on;plot(t(idx),moyenne(idx),'r+');xlabel('temps en secondes')
subplot(212);plot(t,variance,'-+');grid;title('\sigma(r)'); hold
on;plot(t(idx),variance(idx),'r+');xlabel('temps en secondes')
figure
plot(t,y-yhat)
title('y-yhat');xlabel('temps en secondes')

```